University of Ostrava

Institute for Research and Applications of Fuzzy Modeling

Generating a fuzzy rule base with an additive interpretation

Martin Štěpnička and Radek Valášek

Research report No. 55

2004

Submitted/to appear:

16th IFAC World Congress

Supported by:

Grant IAA 1187301 of the GA AV ČR; Deutsche Forschungsgemeinschaft as part of the Collaborative Research Center "Computational Intelligence" (531)

University of Ostrava
Institute for Research and Applications of Fuzzy Modeling
Bráfova 7, 701 03 Ostrava 1, Czech Republic

tel.: +420-59-6160234 fax: +420-59-6120 478
e-mail: Martin.Stepnicka@osu.cz

Abstract

Since 1965 the fuzzy set theory and its application have deep development especially in many disciplines close to the automatic control of processes. A fuzzy model has been shown to be able to approximate the behaviour of many complex processes. Very robust fuzzy controller can be constructed in various ways. One of them, learning algorithm, is focused in this paper while the approximation idea has been brought from the technique called F-transforms.

KEYWORDS: Fuzzy control, Fuzzy models, Rule based-systems, Fuzzy sets, Learning algorithms.

1 Introduction

A new methodology has been brought to the control theory by L.A. Zadeh who introduced a fundamental idea of expressing dependencies between variables by conditional sentences with fuzzy predicates. These conditional sentences called fuzzy rules made possible to use a fragment of human language in control algorithms. An expert knowledge is then represented in the form of fuzzy rules forming a fuzzy rule base (FRB) for some specific control. Such system using a fuzzy rule base is called fuzzy rule based system and its main advantage is a high robustness.

However, there exist such systems to be controlled where the expert knowledge acquisition is not a trivial task or transformation of such knowledge into FRB would be technically hardly feasible. For these cases, and not only for them, algorithms called *learning* are taken into account. Usually, a learning algorithm works with some training data obtained by one or more experiments. These training data serve us as a pattern of behaviour of a modeled system and the chosen learning algorithm transforms them into a FRB for respective control.

Although the learning algorithms provide fuzzy rule bases describing the controlled proces, these fuzzy rule bases are not usually ready for implementation into the respective inference engines. They suffer from complexity, redundancy or inconsistency. To get rid of mentioned problems many sophisticated algorithms have been developed (see [Setnes et al.(1999)]; [Dvořák and Novák(2004)]; [Novák(2001)]).

The motivation of this paper is to provide a new approach to an interpretation of fuzzy rules as well as their construction by learning algorithm using the extended fuzzy transforms (see [Perfilieva(2003)]). This algorithm avoids the problems of inconsistency and redundancy. Furthermore, it provides the user a possibility to increase or decrease the complexity of generated FRB with respect to user's requirements on accuracy.

2 Preliminaries

For the whole paper we restrict our focus just on multiple-input-single-output systems. Let us stress that although a set of multiple-input-single-output systems is not equivalent to a multiple-input-multiple-output system, usually, we are able to construct such systems of the first type that they control the proces sufficiently.

Let us consider a general FRB consisting of n fuzzy rules of the following form

$$\textbf{IF}\ \mathrm{Ant}_i\ \textbf{THEN}\ \mathrm{Cons}_i, \quad 1 \le i \le n. \tag{1}$$

Each type of such fuzzy rule base can be determined by a certain form of consequents Cons_i, while all of them have the same form of the antecedent (see [Perfilieva(1999)]). We can distinguish between three such types of fuzzy rule bases.

- *Singleton* FRB with consequents given by numerical values (fuzzy singletons).
- *Takagi-Sugeno* FRB with consequents given as linear combinations of the input variables appearing in respective antecedents.
- *Linguistic* FRB with consequents given linguistically using fuzzy sets e.g. *very small, more or less big, about five, etc.*

The first two types are sometimes considered to be the only one while the first one is a special case of the second one. Learning algorithm generating these types of FRB are quite deeply investigated. Usually, the fuzzy sets appearing in the antecedents are determined by fuzzy cluster analysis and then the linear combinations for consequents are generated above the constructed fuzzy clusters. The fundamental method is called *c-means* (see [Bezdek(1981)]).

The third type of FRB is, perhaps, the most usual one and we focus on it. Let us consider a multiple-input-single-output system with p input variables. Each rule from such FRB is written in the following form

$$\textbf{IF } x_1 \text{ is } \mathcal{A}_i^1 \textbf{ AND } \cdots \textbf{ AND } x_p \text{ is } \mathcal{A}_i^p$$
$$\textbf{THEN } y \text{ is } \mathcal{F}_i, \tag{2}$$

where $1 \leq i \leq n$ and linguistic expressions $\mathcal{A}_i^1,\ldots,\mathcal{A}_i^p$, $\mathcal{F}_i$ are represented by suitable fuzzy sets $\mathbf{A}_i^1, \ldots, \mathbf{A}_i^p$, $\mathbf{F}_i$.

Each rule (2) can be interpreted as a fuzzy relation with help of fuzzy sets mentioned above. In order to distinguish two main cases of the interpretation of FRB, we follow the notation of I. Perfilieva (see [Perfilieva(1999)]) and write $R_i^c(\mathbf{x}, y)$ if the relation is given as follows

$$R_i^c(\mathbf{x}, y) = \mathbf{A}_i^1(x_1)\mathbf{t}\ldots\mathbf{t}\mathbf{A}_i^m(x_m)\mathbf{t}\mathbf{F}_i(y) \tag{3}$$

and $R_i^d(\mathrm{x}, y)$ if the relation is given as follows

$$R_i^d(\mathbf{x}, y) = \mathbf{A}_i^1(x_1)\mathbf{t}\ldots\mathbf{t}\mathbf{A}_i^m(x_m) \rightarrow_{\mathbf{t}} \mathbf{F}_i(y), \tag{4}$$

where symbols $\mathbf{t}$ and $\rightarrow_{\mathbf{t}}$ mean some t-norm and its adjoint residuum. Fuzzy relations $R_i^c(\mathbf{x}, y)$ and $R_i^d(\mathbf{x}, y)$ are used in the following two fuzzy relations interpreting the given FRB:

$$R^{DNF} = \bigvee_{i=1}^{n} R_i^c(\mathbf{x}, y) \tag{5}$$

or

$$R^{CNF} = \bigwedge_{i=1}^{n} R_i^d(\mathbf{x}, y), \tag{6}$$

while formulas (5) and (6) are called disjunctive normal form and conjunctive normal form, respectively. The choice of concrete t-norm $\mathbf{t}$ specifies the interpretation of the given FRB e.g. with t-norm equal to the min operator we get well known Mamdani-Assilian interpretation.

3 Fuzzy transforms

This section recalls specific technique of an approximate representation of a continuous function. The author of this technique, I. Perfilieva, published a method called fuzzy transforms (in short F-transforms) which is based on two transforms - the direct one and the inverse one (see [Perfilieva(2003)], [**?**]).

3.1 F-transforms for functions with one variable

An interval $[a, b]$ of real numbers will be considered as a common domain of all functions in this subsection.

Definition 1 Let $x_i = a + h(i-1)$ be nodes on $[a, b]$ where $h = (b-a)(n-1)$, $n \geq 2$ and $i = 1, ..., n$. We say that functions $\mathbf{A}_1(x), \ldots, \mathbf{A}_n(x)$ defined on $[a, b]$ are *basic functions* if each of them fulfills the following conditions:

- $\mathbf{A}_i : [a, b] \to [0, 1], \ \mathbf{A}_i(x_i) = 1,$
- $\mathbf{A}_i(x) = 0$ if $x \notin (x_{i-1}, x_{i+1})$ where $x_0 = a, \ x_{n+1} = b,$
- $\mathbf{A}_i(x)$ is continuous,

- $\mathbf{A}_i(x)$ strictly increases on $[x_{i-1}, x_i]$ and strictly decreases on $[x_i, x_{i+1}]$,
- $\sum_{i=1}^{n} \mathbf{A}_i(x) = 1$, for all x,
- $\mathbf{A}_i(x_i - x) = \mathbf{A}_i(x_i + x)$, for all $x \in [0, h],\ i = 2, \ldots, n-1,\ n > 2$,
- $\mathbf{A}_{i+1}(x) = \mathbf{A}_i(x - h)$, for all $x,\ i = 2, \ldots, n-2,\ n > 2$.

We can say that fuzzy sets $\mathbf{A}_i(x)$ determine a uniform fuzzy partition of real interval $[a, b]$. If we avoid requiring the last two conditions, fuzzy sets $\mathbf{A}_i(x)$ determine just a fuzzy partition of $[a, b]$. Each basic function $\mathbf{A}_i(x)$ can be viewed as a fuzzy set *approximately* x_i.

Definition 2 Let $f(x)$ be any continuous function on $[a, b]$ and $\mathbf{A}_1(x), \ldots, \mathbf{A}_n(x)$ be basic functions. We say that the n-tuple of real numbers $[F_1, \ldots, F_n]$ is *the direct F-transform* of f w.r.t. $\mathbf{A}_1(x), \ldots, \mathbf{A}_n(x)$ if

$$F_i = \frac{\int_a^b f(x)A_i(x)dx}{\int_a^b A_i(x)dx}. \tag{7}$$

The direct F-transform transforms a function to a real vector which serves us as its discrete representation. Real *components* F_i given by 7 take into account the whole support of $\mathbf{A}_i(x)$ what means that this technique uses a specific averaging through all the values in a neighbourhood of node x_i. This provides the robustness, for fuzzy techniques so typical, what was used in many application e.g. noise reduction (see [**?**]).

Definition 3 Let $[F_1, \ldots, F_n]$ be a direct F-transform of a function $f(x)$ with respect to $\mathbf{A}_1(x), \ldots, \mathbf{A}_n(x)$. The function

$$f_n^F(x) = \sum_{i=1}^{n} F_i \mathbf{A}_i(x) \tag{8}$$

will be called the *inverse F-transform*.

The inverse F-transform provides an appropriate continuous approximation of the original function. Its form, linear combination of fuzzy sets determining a uniform fuzzy partition, seems to be very useful for further applications.

3.2 Generalization

In order to be able to describe somehow multiple-input-single-output system, we must generalize the technique of F-transform for functions with more variables. This step has been done e.g. in ([**?**]) or in ([Štepnička and Valášek(2004)]). Here, because of lack of space we recall the case of functions with two variables. The idea of this generalization is still the same and the two variables are enough for illustration to give the reader an impression of the main ideas.

Let a rectangle $[a, b] \times [c, d]$ be a common domain of all functions in this subsection. We define a *system of basic functions* as a set of basic functions $\mathbf{A}_1(u), \ldots, \mathbf{A}_n(u)$ determining a uniform fuzzy partition on $[a, b]$ and basic functions $\mathbf{B}_1(v), \ldots, \mathbf{B}_m(v)$ determining a uniform fuzzy partition on $[c, d]$.

Definition 4 Let $f(u, v)$ be a continuous function on $[a, b] \times [c, d]$ and let $\{\mathbf{A}_i, \mathbf{B}_j\}_{\substack{i=1 \\ j=1}}^{n,m}$ be a system of basic functions on $[a, b] \times [c, d]$. Then the matrix $[F_{ij}]$ given as follows

$$F_{ij} = \frac{\int_c^d \int_a^b f(u, v)\mathbf{A}_i(u)\mathbf{B}_j(v)dudv}{\int_c^d \int_a^b \mathbf{A}_i(u)\mathbf{B}_j(v)dudv} \tag{9}$$

will be called the direct F-transform of f w.r.t. the given system of basic functions.

Definition 5 Let $[F_{ij}]$ be the direct F-transform of a function $f(u, v)$. Then the function

$$f_{n,m}^F(u, v) = \sum_{i=1}^{n} \sum_{j=1}^{m} F_{ij} \mathbf{A}_i(u)\mathbf{B}_j(v) \tag{10}$$

will be called the inverse F-transform.

3.3 Data-based model

All the definitions presented in the previous subsections required continuity. Although it is a natural requirement and especially while we model some systems with physical quantities the continuity is necessary because quantities like temperature or pressure cannot provide discontinuous changes, it causes some problems. Especially integral formulas in definitions causes the numerical evaluation too complex.

Furthermore, we usually do not have the full knowledge of a function f but just at some nodes. This situation is again typical even for systems with quantities where only a continuous change is predictable. This is usually caused by the fact that f is known only theoretically and in practice we have a set of measurements (of e.g. temperature)

$$(x_k, f(x_k)) \quad k = 1, \dots, r, \tag{11}$$

where $f(x_k)$ is a value of quantity f measured at node x_k.

In fact, continuous knowledge of $f(x)$ is a limit case of having data (11) as well as integral is a limit summation. Keeping in mind this, we redefine F-transforms for discrete data set.

The direct F-transform of function f known at nodes $\{x_k\}_{i=1}^{r}$ w.r.t. some basic functions $\mathbf{A}_1, \dots, \mathbf{A}_n$ is given as follows

$$F_i = \frac{\sum_{k=1}^{r} f(x_k)\mathbf{A}_i(x_k)}{\sum_{k=1}^{r} \mathbf{A}_i(x)} \quad i = 1, \dots, n. \tag{12}$$

For a function with two variables $f(u, v)$ the formula of the direct F-transform changes analogously. Let us be given data

$$(u_k, v_k, f(u_k, v_k)) \quad k = 1, \dots, r, \tag{13}$$

then the direct F-transform of f w.r.t. a system of basic functions $\{\mathbf{A}_i, \mathbf{B}_j\}_{\substack{i=1\\j=1}}^{n,m}$ is given as follows

$$F_{ij} = \frac{\sum_{k=1}^{r} f(u_k, v_k)\mathbf{A}_i(u_k)\mathbf{B}_j(v_k)}{\sum_{k=1}^{r} \mathbf{A}_i(u_k)\mathbf{B}_j(v_k)}. \tag{14}$$

4 Extended fuzzy transforms

From mathematical point of view, the crucial point of fuzzy control consists in fuzzy relation. In section 2, we have mentioned that a FRB is interpreted by a fuzzy relation. Concrete interpretations are given by formulas (5), (6) and by concrete choice of t-norm $\mathbf{t}$.

Having in mind all the properties of F-transforms and their advantages including low computational complexity and high speed of this algorithm leads to a natural idea of an extension of F-transforms for fuzzy relations. Such extended F-transforms could approximate a fuzzy relation which serves us as an approximate interpretation of some FRB in fuzzy control.

4.1 Fuzzy set-valued function

In fuzzy control, instead of crisp control function $f : X \to Y$, we use a fuzzy relation $R : X \times Y \to [0, 1]$ describing the control function which is in principle unknown.

But the control function cannot be arbitrary and continuity is required. That is why it is not necessary to deal with arbitrary fuzzy relation R and something analogous to the continuity of crisp function is required. Let us restrict the choice of approximated fuzzy relation only to a "continuous" fuzzy set-valued function. This is a natural requirement because, briefly said, fuzzy set-valued function is a mapping which maps each element from its domain to a bounded closed fuzzy number.

Definition 6 Let $\mathcal{F}_0(\mathbb{R})$ denote the set of fuzzy sets $\mathbf{F} : \mathbb{R} \to [0, 1]$ with the following properties:

- $\{x \in \mathbb{R} : \mathbf{F}(x) = 1\} \neq \emptyset$
- $\mathbf{F}_\alpha = \{x \in \mathbb{R} : \mathbf{F} \geq \alpha\}$ is a bounded closed interval in $\mathbb{R}$ for each $\alpha \in (0, 1]$ i.e. $\mathbf{F}_\alpha = [F_\alpha^-, F_\alpha^+]$, where $F_\alpha^- = \inf \mathbf{F}_\alpha, F_\alpha^+ = \sup \mathbf{F}$.

Then $\mathbf{F} \in \mathcal{F}_0(\mathbb{R})$ will be called the *bounded closed fuzzy number* (BCFN).

Definition 7 A mapping $\Phi : X \subset \mathbb{R} \to \mathcal{F}_0(\mathbb{R})$ is a fuzzy relation called *fuzzy set-valued function.*

Definition 8 Let $I(\mathbb{R})$ denote the set of all real bounded closed intervals. Then function

$$\begin{aligned} \varphi &: X \to I(\mathbb{R}), \\ x &\mapsto \varphi(x) = [\varphi^-(x), \varphi^+(x)], \end{aligned} \tag{15}$$

such that $\varphi^-(x), \varphi^+(x)$ are real functions on X will be called the *interval valued function.*

It has been shown (see [Zhang et al.(1998)]) that Φ is a fuzzy set-valued function if and only if the *α-level function*

$$\Phi_\alpha(x) = [\Phi_\alpha^-(x), \Phi_\alpha^+(x)] \tag{16}$$

of $\Phi(x)$ is an interval valued function for each $\alpha \in (0,1]$.

Let us define a continuous fuzzy set-valued function as a fuzzy set valued function such that for each $\alpha \in (0,1] :\ \Phi_\alpha^-(x), \Phi_\alpha^+(x)$ are continuous real functions on X.

For more details see e.g. ([Zhang et al.(1998)]) or ([Zhang and Wang(1998)]).

4.2 Zadeh's Extension

We have recalled F-transforms as a technique of approximate representation of a continuous function. Then we have briefly mentioned fuzzy set-valued functions as a fuzzy relations naturally extending the notion of a real valued function. Now, applying Zadeh's extensional principle to formula (7) defining the direct F-transforms leads to an extended technique for approximate representation of a continuous fuzzy set-valued function.

Definition 9 Let $\Phi : X \to \mathcal{F}_0(Y),\ Y \subset \mathbb{R}$ be a continuous fuzzy function and $\mathbf{A}_1(x), \ldots, \mathbf{A}_n(x)$ be basic functions determining a uniform fuzzy partition of $X \in I(\mathbb{R})$. We say that the n-tuple of fuzzy sets $[\mathbf{F}_1(y), \ldots, \mathbf{F}_n(y)]$ on Y is the *extended direct F-transform* of Φ with respect to $\mathbf{A}_1(x), \ldots, \mathbf{A}_n(x)$ if

$$\mathbf{F}_i(y) = \bigvee_{\substack{\xi \in Y^X \\ \frac{\int_X \xi(x)\mathbf{A}_i(x)dx}{\int_X \mathbf{A}_i(x)dx} = y}} \left(\bigwedge_{x \in X} \Phi(x)(\xi(x)) \right). \tag{17}$$

Because of a combination of supremum and infimum in formula (17) it is not necessary to consider all functions $\xi(x) \in Y^X$. It is sufficient to consider functions $\xi(x)$ which are either equal to $\varphi_\alpha^+(x)$ or to $\varphi_\alpha^-(x)$ for some $\alpha \in (0,1]$.

This simplifies formula (17) and we obtain

$$\mathbf{F}_i(y^+) = \alpha \quad \text{where} \quad y^+ = \frac{\int_X \varphi_\alpha^+(x)\mathbf{A}_i(x)dx}{\int_X \mathbf{A}_i(x)dx}, \tag{18}$$

$$\mathbf{F}_i(y^-) = \alpha \quad \text{where} \quad y^- = \frac{\int_X \varphi_\alpha^-(x)\mathbf{A}_i(x)dx}{\int_X \mathbf{A}_i(x)dx}. \tag{19}$$

Definition 10 Let $\mathbf{F}_1, \ldots, \mathbf{F}_n$ be an extended direct F-transform of a continuous fuzzy set-valued function Φ w.r.t. given basic functions $\mathbf{A}_1, \ldots, \mathbf{A}_n$. Then fuzzy relation

$$\Phi_n^F(x)(y) = \sum_{i=1}^n \mathbf{F}_i(y)\mathbf{A}_i(x) \tag{20}$$

is called the extended inverse F-transform.

The generalization of the extended F-transforms for fuzzy set-valued functions with more variables is straightforward.

5 Learning

This section deals with an idea of generating an appropriate FRB for a control of some process.

5.1 Data-based model

We recall subsection 3.3 where we did not have the full knowledge of an approximated function but we were just given some discrete data. Then the formulas for the direct F-transform have been modified into the shapes given by formulas (12) and (14), respectively.

Following this idea leads to the case when we have the knowledge of fuzzy set-valued function Φ only at some nodes x_k where $k = 1, \dots, r$ i.e. we have a set of data $(x_k, \Phi(x_k)(y))$. Then the extended direct F-transform is given as follows

$$\mathbf{F}_i(y) = \frac{\sum_{i=1}^{r} \Phi(x_k)(y)\mathbf{A}_i(x_k)}{\sum_{i=1}^{r} \mathbf{A}_i(x_k)}. \tag{21}$$

Formula (20) for the extended inverse F-transform is without any change.

In the case of fuzzy set-valued function with two variables we, analogously, require a set of data $(u_k, v_k, \mathit{\Phi}(u_k, v_k))$, $k = 1, \dots, r$ and the formula is modified as follows

$$\mathbf{F}_{ij}(y) = \frac{\sum_{k=1}^{r} \Phi(u_k, v_k)(y)\mathbf{A}_i(u_k)\mathbf{B}_j(v_k)}{\sum_{k=1}^{r} \mathbf{A}_i(u_k)\mathbf{B}_j(v_l)}. \tag{22}$$

The extended inverse F-transform is then evaluated as follows

$$\Phi_{n,m}^{F}(u, v)(y) = \sum_{i=1}^{n}\sum_{j=1}^{m} \mathbf{F}_{ij}(y)\mathbf{A}_i(u)\mathbf{B}_j(v). \tag{23}$$

The data given in mentioned shape i.e. a collection of crisp values u_k, v_k and fuzzy sets at these nodes $\Phi(u_k, v_k)$ are quite natural. These data may be obtained by questioning some expert. For example, in fuzzy control of a dynamic robot moving through a corridor, we know a distance u between the robot and the middle of the corridor at each moment and we know its change v per time. We ask some expert about a control action. Answer could be either a linguistic expression or a fuzzy number.

5.2 Learning algorithm

Although the approach introduced above is a natural way how to implement the extended F-transforms to turn a collection of data to a fuzzy relation (i.e. FRB), it requires a knowledge of fuzzy sets $\Phi(u_k, v_k)$ i.e. an expert estimates. In order to obtain a learning algorithm generating an FRB from a measured data while the robot is controlled manually we must consider just crisp data.

So, let us be given data (u_k, v_k, y_k) where y_k is a measured control action at node u_k, v_k. The only one difference is that, at first, we somehow transform data y_k to fuzzy sets $\Phi(u_k, v_k)$. We knowingly avoid the usage of word "fuzzification", because it is misleading in this case. This word is intuitively understood as something what changes crispness into fuzziness, but we are not not artificially creating something fuzzy from crisp data. We are, in fact, reconstructing fuzzy data which are, unfortunately, by a sensor or measuring instrument badly reduced to crisp values, but surely measured distance u_k is not precisely the distance between the dynamic robot and the middle of the corridor. Then v_k could not be computed precisely neither. And that is why it is just an illusion to consider values y_k as precise values of the control action.

Finally, let us introduce the whole algorithm from discrete data to an interpretation of the generated FRB step by step. Because of simplicity we consider single-input-single-output system.

1. Measuring training data (x_k, y_k) while y_k is the k-th control action and $k = 1, \dots, r$.
2. Reconstructing the fuzziness of values of y_k. We get data $(x_k, \Phi(x_k))$.
3. Constructing basic functions $\mathbf{A}_i$, $i = 1, \dots, n$. (The choice of shape of basic functions and their number n is an expert decision)

4. Constructing components $\mathbf{F}_i$ of the direct F-transform according to formula (21) w.r.t. the chosen basic functions.

5. Generating an FRB:

$$\textbf{IF } x \text{ is } \mathcal{A}_i \textbf{ THEN } y \text{ is } \mathcal{F}_i,$$

   where $i = 1, \ldots, n$.

6. Interpretation of the previous FRB is given by (20) what can be rewritten into the following **additive** form:

$$\bigoplus_{i=1}^{n} (\mathbf{A}_i \odot \mathbf{F}_i). \tag{24}$$

7. Using an appropriate defuzzification method. Having in mind the shapes of used fuzzy sets, the method *COG* (*Center of Gravity*) is such defuzzification giving good results.

Now, let us briefly remind the basic algorithm used for generating an FRB with the conjunctive interpretation and the usage of linguistic expressions derived from basic trichotomy.

1. Measuring training data (x_k, y_k) while y_k is the k-th control action and $k = 1, \ldots, r$.

2. Reconstructing the fuzziness of values of y_k as well as of values x_k. We get fuzzy data $(\mathbf{X}_k, \mathbf{Y}_k)$.

3. Generating an initial FRB:

$$\textbf{IF } x \text{ is } \mathcal{X}_k \textbf{ THEN } y \text{ is } \mathcal{Y}_k,$$

   where $k = 1, \ldots, r$.

4. Interpretation of the previous FRB is given by the following **conjunctive** form:

$$\bigwedge_{k=1}^{r} (\mathbf{X}_k \rightarrow \mathbf{Y}_k). \tag{25}$$

   Because the number of measured pairs (x_k, y_k) is usually very high (at least hundreds), it is almost impossible to use the initial FRB in practice. The complexity must be reduced, the inconsistency removed and redundancy decreased.

5. Using some sophisticated algorithm for inconsistency elimination (see [Dvořák and Novák(2004)]; [Novák(2001)]).

6. Using some sophisticated algorithm for redundancy analysis and decrease (see [Bělohlávek and Novák(2002)]).

7. Using an appropriate defuzzification method w.r.t. the shapes of used fuzzy sets. Method *DEE* (*Defuzzification of Evaluating Expressions* see [Bělohlávek and Novák(2002)]) is such defuzzification giving good results for linguistically given fuzzy sets like *very small* or *roughly big*. Method *COG* is suitable for the case of the usage of fuzzy numbers.

Let us stress that the last three steps can be avoided (at least partially), but it requires similar analysis of data (x_k, y_k). Some statistical techniques, clustering, averaging etc. can get rid of data causing the redundancy and the inconsistency and reduce the complexity as well.

6 Conclusion

The paper deal with a fuzzy rule base which is generated from given data. The algorithm providing such results is called learning algorithm and has been many times successfully applied in practice. We came to the problem of a construction of an FRB from data from the approximation problem. Having in mind all to of applications and advantages of F-transforms as an approximating method lead us to its generalization. What means that we defined so called extended fuzzy transforms for approximate representation of a continuous fuzzy set valued function. Discretization of such method provides, in fact, an algorithm for data processing returning a fuzzy relation describing a process we controlled while obtaining data. Finally, we have constructed an FRB with the additive interpretation which corresponds to a fuzzy relation given by extended F-transforms.

7 Acknowledgement

This investigation has been partially supported by grant IAA 1187301 of the GA AV ČR and by the Deutsche Forschungsgemeinschaft as part of the Collaborative Research Center "Computational Intelligence" (531).

References

[Bělohlávek and Novák(2002)] Bělohlávek, R., V. Novák(2002). Learning rule base in linguistic expert systems. *Soft Computing.* 7, 79-88.

[Bezdek(1981)] Bezdek, J.C.(1981). *Pattern Recognition with Fuzzy Objective Function Algorithms.* Plenum Press, New York.

[Dvořák and Novák(2004)] Dvořak, A., V. Novák(2004). Formal Theories and Linguistic Descriptions. *Fuzzy Sets and Systems.* 143, 169-188.

[Novák(2001)] Novák, V.(2001). Antonyms and Linguistic Quantifiers in Fuzzy Logic. *Fuzzy Sets and Systems.* 124, 335-351.

[Novák et al.(1999)] Novák, V., I. Perfilieva and J. Močkoř(1999). *Mathematical Prinicples of Fuzzy Logic.* Kluwer, Boston, Dordrecht, London.

[Perfilieva(2003)] Perfilieva, I.(2003). Fuzzy approach to solution of differential equations with imprecise data: application to reef growth problem. In: *Fuzzy Logic in Geology* (R.V. Demicco and G.J. Klir, Ed.). Chap. 9, pp. 275-300. Academic Press, Amsterdam.

[Perfilieva(1999)] Perfilieva, I.(1999). Fuzzy logic normal forms for control law representation. In: *Fuzzy Algorithms for Control* (H.B. Verbruggen and H.-J. Zimmermann and R. Babuška, Ed.). Chap. 5, pp. 111-125. Kluwer, Boston, Dordrecht, London.

[Setnes et al.(1999)] Setnes, M., V. Lacrose and A. Titli(1999). Complexity reduction methods for fuzzy systems. In: *Fuzzy Algorithms for Control* (H.B. Verbruggen and H.-J. Zimmermann and R. Babuška, Ed.). Chap. 8, pp. 185-218. Kluwer, Boston, Dordrecht, London.

[Štepnička and Valášek(2004)] Štepnička, M., V. Valášek(2004). Fuzzy transforms and their application on wave equation. *Journal of Electrical Engineering*, to appear.

[Zhang et al.(1998)] Zhang, Y., G. Wang and S. Liu(1998). Frequency domain methods for the solutions of N-order fuzzy differential equations. *Fuzzy Sets and Systems*, 94, 45-59.

[Zhang and Wang(1998)] Zhang, Y., G. Wang (1998). Time domain methods for the solutions of N-order fuzzy differential equations. *Fuzzy Sets and Systems*, 94, 77-92.